In the Claims:

Cancel claims 3 - 10

4

~~12.~~ (Amended) [A computer program product that implements an execution stack that stores frames for functions written in a plurality of programming languages,] A computer readable medium including computer program code for implementing an execution stack, the computer readable medium comprising:

computer code that stores a first frame on the execution stack for a first function, the first function being written in a first programming language; and

computer code that, in response to the first function calling a second function written in a second programming language, stores a data block on the execution stack before a second frame for the second function, the data block including at least one pointer to a previous frame on the execution stack for a previous function written in the second programming language. [and

a computer readable medium that stores the computer codes.]

5 4

~~13.~~ (Amended) The computer [program product] readable medium of claim ~~12~~, wherein the computer readable medium is selected from the group consisting of CD-ROM, floppy disk, tape, flash memory, system memory, hard drive, and data signal embodied in a carrier wave.

a1

6

~~14.~~ (Amended) A computer system [for implementing] having an execution stack that stores frames for functions written in a plurality of programming languages, the computer system comprising:

a processor;

a memory coupled to the processor that stores the execution stack; and

a computer program operating on the processor that stores a first frame on the execution stack for a first function, the first function being written in a first programming language and, in

31

response to the first function calling a second function written in a second programming language, stores a data block on the execution stack before a second frame for the second function, the data block including at least one pointer to a previous frame on the execution stack for a previous function written in the second programming language.

7
~~15.~~ (Amended) In a computer system, a method for [implementing an execution stack that stores] storing frames for functions written in a plurality of programming languages on an execution stack, the method comprising:

storing a first frame on the execution stack for a first function, the first function being written in a first programming language; and

in response to the first function calling a second function written in a second programming language, storing in local storage at least one pointer to the first frame on the execution stack and storing a second frame on the execution stack for the second function.

25
~~33.~~ (Amended) [A computer program product that implements an execution stack that stores frames for functions written in a plurality of programming languages,] A computer readable medium comprising:

computer code that stores a first frame on the execution stack for a first function, the first function being written in a first programming language; and

computer code that, in response to the first function calling a second function written in a second programming language, stores in local storage at least one pointer to the first frame on the execution stack and stores a second frame on the execution stack for the second function. [; and

a computer readable medium that stores the computer codes.]

26
~~34.~~ (Amended) The computer [program product] readable medium of claim 25 ~~33~~, wherein the computer readable medium is selected from the group consisting of CD-ROM,

floppy disk, tape, flash memory, system memory, hard drive, and data signal embodied in a carrier wave.

27
~~35.~~ (Amended) A computer system [for implementing an execution stack that stores frames for functions written in a plurality of programming languages,] comprising:

a processor;

a memory coupled to the processor that stores [the] an execution stack; and

[an] a computer program operating on the processor that stores a first frame on the execution stack for a first function, the first function being written in a first programming language; and, in response to the first function calling a second function written in a second programming language, stores in local storage at least one pointer to the first frame on the execution stack and stores a second frame on the execution stack for the second function.

a2

28
~~36.~~ (Amended) A data structure stored by a computer readable medium [for implementing an execution stack,] comprising:

a first frame stored by the computer readable medium on [the] an execution stack, the first frame being for a first function written in a first programming language;

a second frame stored by the computer readable medium on the execution stack above the first frame, the second frame being for a second function written in a second programming language; and

a data block stored by the computer readable medium on the execution stack above the second frame, the data block including at least one pointer to the first frame on the execution stack.

Add the following newly drafted claims:

33

34
42. In a computer system having an execution stack that stores frames for functions written in a plurality of programming languages, a method for operating the computer system by utilizing the execution stack, the method comprising:

storing a first frame on the execution stack for a first function, the first function being written in a first programming language; and

in response to the first function calling a second function written in a second programming language, storing a data block on the execution stack before a second frame for the second function, the data block including at least one pointer to a previous frame on the execution stack for a previous function written in the second programming language.

35 34
43. The method of claim 42, wherein the at least one pointer includes a previous stack pointer and frame pointer.

A3

36 34
44. The method of claim 42, further comprising in response to the first function calling the second function, allocating resources for functions written in programming languages other than the second programming language that may be called by the second function.

37 36
45. The method of claim 44, further comprising upon exiting the second function, deallocating the resources for functions written in programming languages other than the second programming language.

38 34
46. The method of claim 42, further comprising catching an exception that was raised during execution of the second function that was not handled by an exception handler for the second function.

39 38
47. The method of claim 46, further comprising identifying an exception handler for the data block to handle the exception and jumping to the identified exception handler.

34